Thesis Project Portfolio

Improving the User Experience of Proof Assistants: A Comprehensive Study of Interface Design and Accessibility

(Techncial Report)

The Mistrust of Formal Proofs in Pure Mathematics

(STS Research Paper)

An Undergraduate Thesis

Presented to The Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Jamie Fulford

Spring, 2025

Department of Computer Science

Table of Contents

Executive Summary

The increasing reliance on digital systems for both mathematical knowledge and software infrastructure creates a fundamental tension between formal verification and intuitive understanding. Both my technical and social research projects examine aspects of this tension from complementary perspectives. In my technical work, I formalized the EPIC calculus in the Rocq proof assistant to provide rigorous guarantees for automatically parallelizing programs with external calls, particularly to large language models. This formalization addresses the critical need for software correctness in systems where manual parallelization is error-prone but formal verification can establish that automatic parallelization preserves program semantics. My STS research complements this technical work by investigating why mathematicians resist adopting formal verification tools like proof assistants despite their potential to enhance mathematical certainty. Together, these projects illuminate a common challenge: as our digital infrastructure grows increasingly complex, formal verification becomes simultaneously more necessary for reliability and more difficult to integrate with human cognitive practices. This challenge matters not only academically but practically, as failures in critical software systems can lead to catastrophic consequences—from the Therac-25 radiation therapy machine that fatally overdosed patients to the Ariane 5 rocket's self-destruction that resulted in a $370 million loss. By examining both the technical development of verification tools and the social resistance to their adoption, my research contributes to understanding how we might bridge the gap between formal verification's theoretical potential and its practical implementation in both mathematical and computational contexts.

My technical research addressed the challenge of formalizing EPIC, a lambda calculus designed for automatic parallelization of external calls in scripting languages. Using the Rocq proof assistant, I developed a formal model of EPIC's syntax and semantics through mutually recursive inductive types and small-step reduction relations. The formalization focused on proving two essential properties: confluence, which ensures that different evaluation orders lead to equivalent results, and well-formedness preservation, which guarantees that evaluation maintains proper variable scoping regardless of execution order. I employed custom mutual induction principles and

techniques for reasoning about nondeterminism to establish these properties, providing a mathematical foundation for EPIC's claim that automatic parallelization preserves program semantics. While time constraints prevented a complete proof of full confluence, the partial results demonstrate the soundness of EPIC's approach to parallelization for deterministic code.

My STS research investigated the persistent mistrust of proof assistants within the mathematical community despite their potential to enhance certainty in both mathematical theorems and software systems. Through historical case studies, firsthand accounts, personal experience with Rocq, and the Social Construction of Technology framework, I examined barriers preventing wider adoption of formal verification tools. The evidence revealed that resistance stems not from technological conservatism but from fundamental misalignments between mathematical cognition and formal verification approaches. Mathematicians struggle with translating intuitive reasoning into machine-verifiable syntax, encounter procedural barriers in tactical proof construction, and find that formalization disrupts the conceptual clarity they value in traditional proofs. My SCOT analysis identified distinct social groups—traditional mathematicians, computer scientists, verification specialists, and younger mathematicians—with different understandings of what proof assistants are and what role they should play. These findings demonstrate that bridging the gap between mathematical practice and proof assistants requires not just technical improvements but social interventions that acknowledge the value of both intuitive insight and machine-checked verification.

Both projects contribute to addressing the gap between formal verification's theoretical power and practical adoption by examining complementary aspects of the problem. My technical formalization provides a foundation for automatic parallelization with mathematical guarantees, while my STS research illuminates the social barriers to wider adoption of verification tools. Future research should focus on developing proof assistants that better accommodate mathematical intuition and cognition, creating interfaces that preserve conceptual clarity while leveraging formal rigor. Additionally, fostering genuine dialogue between mathematicians and computer scientists could help reconceptualize proof assistants as complementary to traditional mathematical reasoning rather than competing approaches.

I would like to express my sincere gratitude to the research team at the University of Pennsylvania, particularly Steve Zdancewic, Stephen Mell, and Joey Velez-Ginorio, for their guidance and collaboration throughout the technical project. I am also deeply grateful to Professor Caitlin D. Wylie from the Department of Engineering and Society for her mentorship and insightful feedback throughout the development of my STS research. Her guidance helped me approach the social dimensions of technological adoption with greater nuance and clarity.